HAL open science

Vehicle recognition and tracking using a generic multi-sensor and multi-algorithm fusion approach

Fawzi Nashashibi, Ayoub Khammari, Claude Laurgeau

▶ To cite this version:

Fawzi Nashashibi, Ayoub Khammari, Claude Laurgeau. Vehicle recognition and tracking using a generic multi-sensor and multi-algorithm fusion approach. International Journal of Vehicle Autonomous Systems, 2008, Special Issue on "Intelligent Vehicles", 6 (1/2), pp.134-154. hal-00922518

HAL Id: hal-00922518
https://inria.hal.science/hal-00922518

Submitted on 27 Dec 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Vehicle recognition and tracking using a generic multisensor and multialgorithm fusion approach

Fawzi Nashashibi,* Ayoub Khammari and Claude Laurgeau

Robotics Centre – École des Mines de Paris,
60 Boulevard Saint-Michel,
Paris Cedex 06, 75 272 France
Fax: +33-1-43-26-10-51
E-mail: fawzi.nashashibi@ensmp.fr
E-mail: khammari@ensmp.fr
E-mail: claude.Laurgeau@ensmp.fr
*Corresponding author

Abstract: This paper tackles the problem of improving the robustness of vehicle detection for Adaptive Cruise Control (ACC) applications. Our approach is based on a multisensor and a multialgorithms data fusion for vehicle detection and recognition. Our architecture combines two sensors: a frontal camera and a laser scanner. The improvement of the robustness stems from two aspects. First, we addressed the vision-based detection by developing an original approach based on fine gradient analysis, enhanced with a genetic AdaBoost-based algorithm for vehicle recognition. Then, we use the theory of evidence as a fusion framework to combine confidence levels delivered by the algorithms in order to improve the classification 'vehicle versus non-vehicle'. The final architecture of the system is very modular, generic and flexible in that it could be used for other detection applications or using other sensors or algorithms providing the same outputs. The system was successfully implemented on a prototype vehicle and was evaluated under real conditions and over various multisensor databases and various test scenarios, illustrating very good performances.

Keywords: intelligent transportation systems; adaptive cruise control; ACC; vehicle detection; vision; laser scanner; object recognition; AdaBoost; fusion; theory of evidence; transferable belief model; TBM.

Reference to this paper should be made as follows: Nashashibi, F., Khammari, A. and Laurgeau, C. (2008) 'Vehicle recognition and tracking using a generic multisensor and multialgorithm fusion approach', *Int. J. Vehicle Autonomous Systems*, Vol. 6, Nos. 1/2, pp.134–154.

Biographical notes: Fawzi Nashashibi is an MSc in Automation, Industrial Engineering and Signal Processing. He received a PhD in Robotics from Toulouse University (LAAS/CNRS – France). He received the HDR title (accreditation to supervise researches) from the University of Jussieu (Paris 6). Since 1994, he is a Senior Researcher in the robotics centre of the École des Mines de Paris. He was previously a Research Engineer at PROMIP and a Technical Manager at Light Co.

Ayoub Khammari received a PhD in Computer Science from the École des Mines de Paris in February 2006. Currently, he is a Research Engineer at the Robotics centre of the École des Mines de Paris. His current research interests include object detection, recognition and tracking using image analysis.

Copyright © 2008 Inderscience Enterprises Ltd.

Claude Laurgeau is a Professor of Transport Robotics at the École des Mines de Paris, Founder and Head of the Robotics Research Centre and Former Head of the Robotics Research at the Informatics Agency (1982–1987). He contributed to create several startups in the domain or robotics. His teaching areas include robotics, real-time informatics, image processing and automation. Education: IAE Engineer (1979), PhD and Docteur d'État (University of Nantes, 1968). He also received several national and international distinctions.

1 Introduction

Intelligent driver assistance is an area of active research among automotive manufacturers, suppliers and universities with the aim of reducing injury and accident severity. According to French road security statistics, about 28% of corporal accidents result from rear or chain collisions. These are mainly due to the misevaluation of the security distance by the driver, late or inadequate braking or a high inappropriate speed. Adaptive Cruise Control (ACC) systems aim to help the driver to ensure that there is enough distance to the car ahead, even if it unexpectedly lowers the speed and can even automatically adjust acceleration and braking manoeuvres.

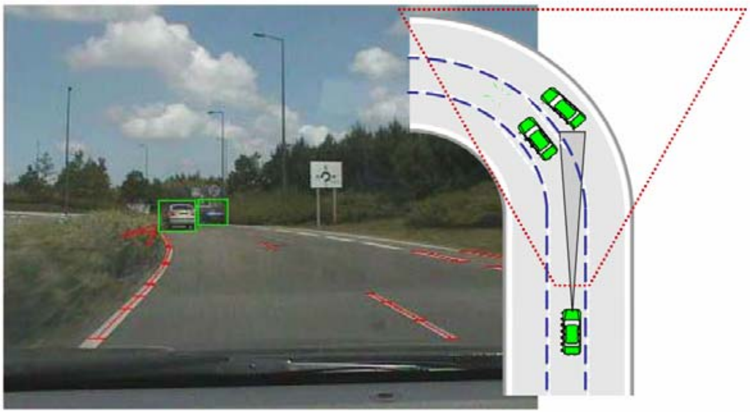
ACC requires three main tasks: perception, localisation and risk assessment. The perception task consists of detecting all the vehicles ahead. The localisation task deals with the positioning of the detected obstacles on the road and therefore the identification of the main target. The risk assessment consists then in elaborating a risk indicator based on the principal obstacle's distance and speed, as well as the host vehicle's speed and, if available, information about the adherence, the braking capacities, the driver behaviour, the visibility, etc.

The first ACC system was developed in 1995 by Nissan on the model *Diamente*. Since then, many other car manufacturers have offered such systems as an option mainly for their luxury brands. Today's ACC systems on the market are mostly based on a radar technology that provides immediately the distance and the speed of obstacles ahead.

While such systems perform well on highways and are not sensitive to the daytime or rainy weather conditions, they suffer from some shortcomings, namely their limited field of view (about 16°) and their low lateral resolution. This could lead to a bad positioning of the targets on the road therefore a misidentification of the right car ahead as shown as an example in Figure 1. Moreover, it gives no reliable information to separate vehicles from other obstacles, which can lead to some false detections.

For these reasons, recent researches were conducted to improve the current ACC systems by using or adding new sensors like cameras or laser scanners. For example, in 2000, researchers from Daimler Chrysler (Gern et al., 2000) used a camera to correct the lateral position of radar targets based on symmetry. A similar strategy was adapted in Fang et al. (2001) but using stereovision. In this approach, vision is only used to enhance radar targets and is not involved in the detection's process itself. In 2001, Steux (2001) proposed an approach that combines both, images and radar clues to form targets, based on a Bayesian network. Recently, the firm Mobileye proposed an ACC system (Stein et al., 2003) based only on monocular vision. This can seem a bit pretentious but according to the authors, they were able to achieve a precision that is comparable with that of radar by estimating the ego-motion of the host vehicle.

Figure 1 A typical case where radar fails to identify the main target



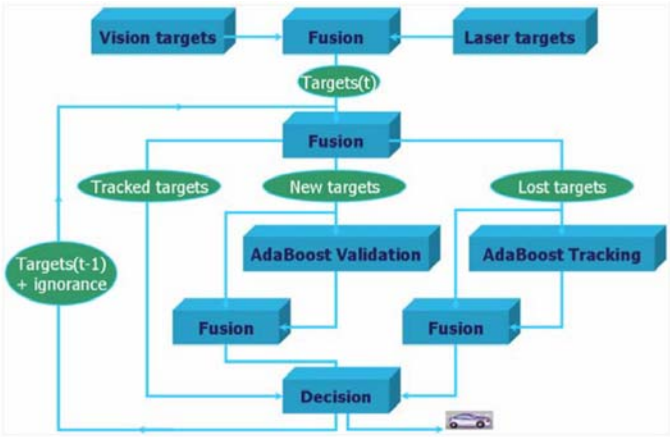
Alternatively, other researches were conducted by using a 2D or 3D laser scanner instead of the radar. The main motivation lies in the wide field of view of a laser scanner compared to radar. For instance, one can refer to Trassoudaine et al. (2004), where a 2D laser scanner is used to generate targets hypothesis and stereovision to validate obstacles on the road. In the same reference is presented a system using a 3D laser scanner to initialise targets' tracks and monocular vision for tracking them between two laser frames (4 Hz). On can argue that such solutions are expensive given the cost of a laser scanner. But recently, affordable lidars have come to the market and their prices are still decreasing from year to year.

For our system (Khammari, 2006), we have chosen a solution that combines monocular vision and a 2D laser scanner for the reasons cited above. One of the originalities of our approach lies in the tight cooperation between these two sensors for the detection task, generating both target hypotheses and detection confidence levels.

2 System overview

In this paper, we will mainly address the perception and the fusion tasks with a focus on the improvement of its robustness. We will also talk briefly about the approach we use for the localisation task. The diagram on Figure 2 gives an overview of the perception system. In order to understand the strategy used for this task, we have to introduce two notions commonly used to assess the detection's performance. These are: the Detection Rate (DR) and the False Alarm Rate (FAR). The DR represents the ratio of detected targets among target that are really present in the scene. The FAR represents the ratio of the false positives among the detected targets. The ultimate goal will be then both to maximise the DR and to minimise the FAR. This is why our system is based on two modules which are the vehicle hypothesis generation and the validation step. Each processing relative to a given sensor or algorithm applied on a target returns a confidence level ranging between 0 and 1 about the real presence of a vehicle. We have chosen the Transferable Belief Model (TBM) of the theory of evidence to handle and combine these confidence levels. This choice will be justified in Section 5.

Figure 2 System's architecture



One can notice here that fusion is used only for the classification purpose and not for the estimation of the target movements that calls usually upon prediction methods like Kalman filters, particle filters, etc. In fact, we believe that we have to better check exhaustively the presence or not of a vehicle before estimating its movement. However, we are convinced that the estimation task is as important as the classification one if we want to build a complete and robust ACC system, since it gives not only a better temporal understanding of the scene but also filtered dynamic parameters of the obstacles for the risk assessment.

At each iteration, the current targets Targets(t) that are extracted separately from the image and from 2D laser data are matched and fused. The fusion between Targets(t) and Targets(t – 1) leads to three target categories that will be processed differently:

- *New targets*: targets of the frame t that have not been matched with any target of the frame t – 1, such as typically passing vehicles or very distant ones but can also be a late detection. These targets need to be classified at least once. This is why they are processed through the *AdaBoost Validation* module. On can notice that the classification score, considered as the classification confidence will be fused with the current target's confidence. That's how confidence is propagated.
- *Tracked targets*: targets of the frame t that have been matched with at least one target of the frame t – 1. It is the most common case. Such targets go straightforward to the decision module.
- *Lost targets*: targets of the frame t – 1 that have not been retrieved in the frame t. These targets need to be tracked. They are processed through the *AdaBoost Tracking* module. Confidences are propagated the same way as for the new targets category.

The final confidences will be used for the decision making. This module, as well as the ignorance handling will be detailed in the section dedicated to fusion.

In the following sections, we will briefly present each component of this diagram. Firstly, the vehicle hypothesis generation using both monocular vision and the 2D laser scanner will be addressed. Then, we will describe the validation step based on the classification algorithm called 'AdaBoost/GA' used for the recognition and tracking processes. The generic fusion framework for combining the confidences from our two sensors and the validation step algorithms will be presented. We will then describe in few words the localisation step. Before concluding, we will show some results from off-line and online experiments illustrating the overall system performance under different traffic, lightening and environment conditions.

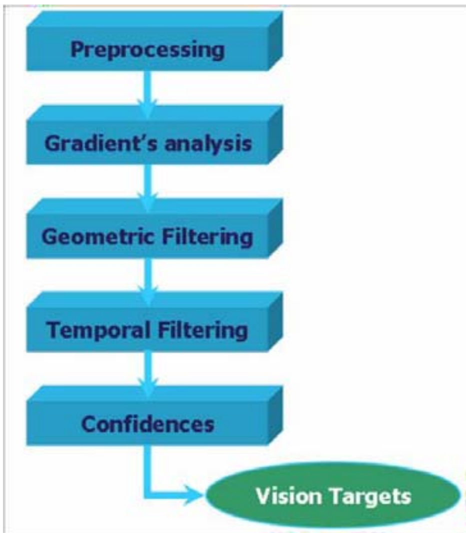
3 Vehicle hypothesis generation

The goal of this module is to initialise the detection system with possible vehicles locations as soon as they appear in the scene. The first effect of this is to maximise the DR. At this level, it does not matter yet if it generates false detections. The FAR minimisation will be addressed in the next section. However, this module must run very quickly since it is executed at each frame and explores a relatively large region of interest. We will describe in the following the way we generate vehicle candidates using vision and laser telemetry.

3.1 Vision-based target generation

Many image clues such us texture, edges, shadows, symmetry, colour, etc. were used in literature to detect vehicles. A good description of these clues can be found in Kalinke et al. (1998). Among them, shadows underneath vehicles are considered as one of the most significant clues indicating an obstacle presence. In our approach, we tried to find a more general clue based on the negative vertical gradient, due to shadows, wheels and bumpers found in the bottom rear view of a vehicle. This will be the main clue used to generate vehicle candidates. We preferred a single clue approach to a saliency map one because it is far less time consuming. Figure 3 gives an overview of the whole vision-based target generation process.

Figure 3 Vision-based targets generation steps



Firstly, we define a region corresponding to 5 – 100 m inside which we limit the search. Besides, to reduce the computation time due to the scene investigation, we start by applying a three-level Gaussian pyramid filter as described in Sun et al. (2002). The top of the image (distant vehicles) is processed at the second level of the pyramid and the image's bottom (near and mid distant vehicles) is processed at the third level. Another advantage of using the Gaussian pyramid is to keep only the most salient structural features so that candidate vehicle locations are easier to get.

The key step of the detection process is the gradient's analysis. It aims to detect local gradient maxima which will help us locate vehicle candidates. All the next operations depend on its results. If this step misses maxima, it will be difficult to find it in the next steps. Therefore, particular attention must be given to this point. Firstly, the negative vertical gradient image is calculated using a biased Sobel 3 × 3 operator (see Figure 4) that delivers thinner edges; then, local maxima are extracted. Thus, we have developed a special adaptive threshold that operates on the gradient image. The threshold value is given by the formula in Figure 4. Pixels with a gradient intensity higher than the threshold are then retained for gradient maxima as shown on Figure 5.

Figure 4 Gradient masks and the adaptive threshold formula

$$\begin{pmatrix} -1 & -2 & -1 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{pmatrix} s_k(i,j) = \frac{(I \oplus V_k) + (I \ominus V_k)}{2}(i,j)$$

Figure 5 Gradient's analysis process: (a) source image, (b) gradient image and (c) maxima

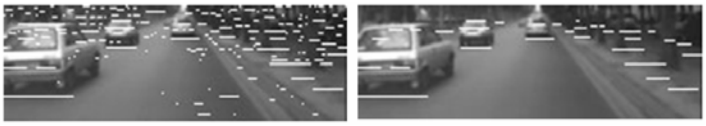
(a)

(b)

(c)

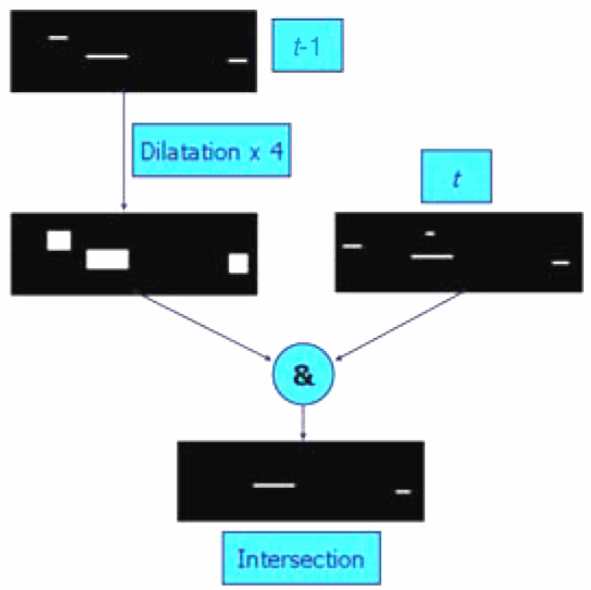
The binary image is then labelled. For each object, we extract the longest horizontal segment. It should be noted that due to the complexity of the scenes, some false maxima are expected to be found. We use some heuristics and constraints to suppress them using perspective projection constraints under the assumption of a flat road. This is what we call 'geometric filtering'. It consists in eliminating all maxima if their 3D position is outside what we consider as the road zone of interest. This zone or corridor is delimitated by our trajectory or path and the estimation of the current road width. An example is given in Figure 6.

Figure 6 Geometric filtering process: before and after filtering



The idea behind the temporal filtering step is to eliminate some basic false detections due to road irregularities or inert shadows. This will make the work of the validation module easier and less time-consuming. For this sake, we evaluate the temporal presence of each binary object as described in Figure 7.

Figure 7 Temporal filtering process



3.2 Laser-based target generation

The laser 2D and 3D rangefinders constitute an ideal sensor for many applications and in particular for obstacles detection. The lidar we use is an 'IBEO LD Automotive' time of flight laser sensor that covers a wide area of 270° with a 0.25° angular resolution. The literature abounds of techniques employing this type of sensor primarily for SLAM-like applications (Früh and Zakhor, 2001; Moutarlier and Chatila, 1989; Nashashibi and Devy, 1993; Zhao and Shibasaki, 2003). For a few years, new applications deal with 3D modelling of urban environments (Ammoun et al., 2004; Hancock et al., 1998; Ye and Borenstein, 2002) and moving obstacle detection (Baltzakis et al., 2003; Gavrila et al., 2001; Mendes and Nunes, 2004). Recent European projects (CARSENSE, PROTECTOR, SAVE-U) and others (e.g. programme PATH) are devoted to the design of multisensor systems dedicated to obstacle detection for applications related to road safety (Langheim et al., 2001); those applications exploit mainly telemetric sensors, vision and/or one or multiple radars (Fanping and Ching-Yao, 2005; Newman, 1999; Victorino et al., 2003).

We use a segmentation-based technique in order to generate targets from laser data. A template-based matching technique is also exploited in order to identify vehicle-like targets and separate them from other obstacles on the road. Each laser scan is composed of 1080 range data. A polygonal segmentation has been performed to each profile. The segments are then analysed and classified accounting their respective lengths and the number of raw data they contain, their distance from the sensor and their angular visibility in their neighbourhoods. We also calculate uncertainty on the segments by propagating data imprecision. Then, segments are filtered, fused or cut considering fuzzy criteria. The study of the configurations obtained (adjacency graph) makes it possible to gather neighbouring segments in the given predefined objects or classes (car, motorcycle, truck, undetermined). A level of confidence about the classification is also provided but it evolves with time. Lastly, *a* tracking is assured in order to: confirm the presence of each target, make evolve/move its class, deliver a level of confidence and to estimate its state vector (position, orientation, speed) (Figure 8).

Figure 8 Laser-based targets generation steps

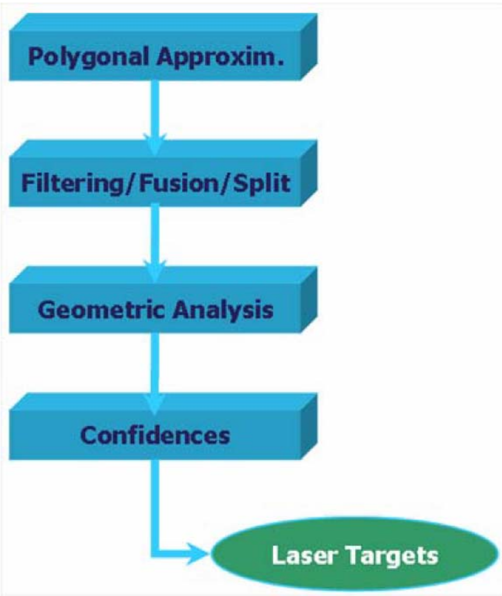
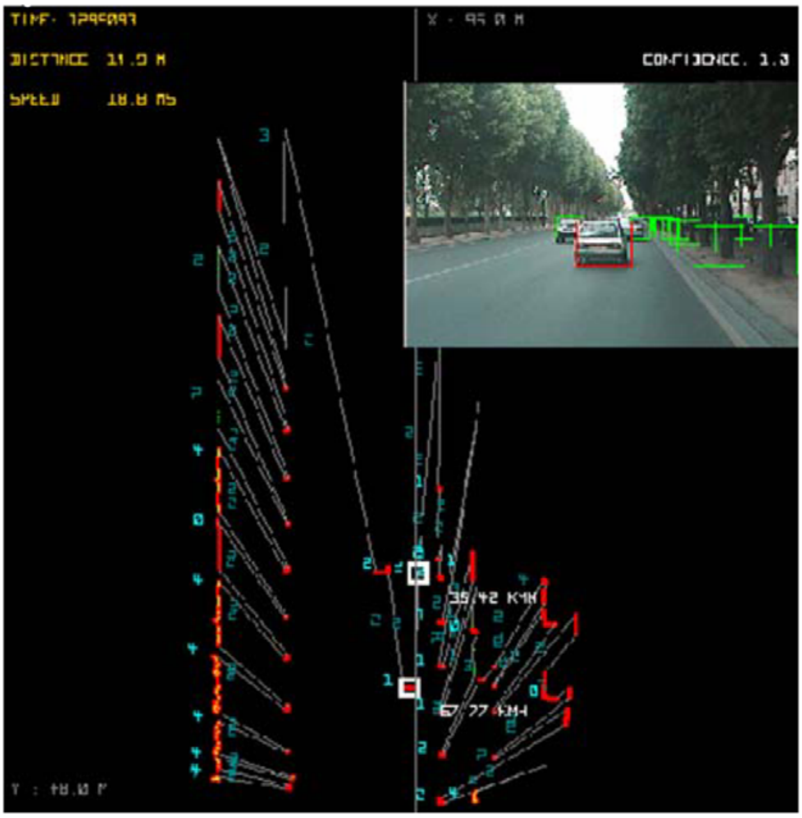


Figure 4 illustrates the stand-alone functioning of laser-based detection (without vision). The framed targets represent the targets located on our lane, their colours represent their respective levels of confidence (ranging from white: 1 to black: 0). Respective absolute speeds of the targets are posted in the vicinity using the same colour whereas the classes of the targets are displayed in Cyan (1: VL, 2: 2R, 3: PL, 4: undetermined). Red segments are valid targets. In yellow is the information about the distance to the nearest lane target, the host vehicle speed and the elapsed time (Figure 9).

A problem remained nevertheless, that of the distinction between two adjacent vehicles moving side by side on two distinct lanes. In that particular case, laser data segmentation would return a unique wide segment wrongly assuming a grouped obstacle. In order to solve this problem, Kalman filtering-based tracking was a good solution. However, a second complementary sensor can prove to be necessary. Thus, an algorithm-based on true pattern recognition can make it possible to separate the two targets even easier. Only collaboration between the image processing and the laser data thus makes it possible to distinguish the two targets.

Figure 9 A bird view of laser based targets detection (for colours see online version)



4 Vision-based vehicle validation

The goal of this module is to give a classification score to each of the vehicle candidates generated by the previous module. The idea is to minimise the FAR. As described in Section 1, two main algorithms will be used here: the single frame validation (classification) and the multiple frame validation (the tracking). Both algorithms are based on a classification algorithm that will be discussed in the next paragraphs. Note that this module should run in real time. This justifies why some smart accelerations were adopted.

4.1 Classification algorithm

Verifying a hypothesis is essentially a two-class pattern classification problem: vehicle versus non-vehicle. To deal with this problem, the most common method found in literature is based on wavelet transforms for features extraction and SVM for classification (Avidan, 2003; Sun et al., 2000). Another classification method, AdaBoost, described in Freund and Schapire (1997) showed satisfactory results in the case of pedestrian detection (Viola and Jones, 2001) and classification (Abramson and Steux, 2004). The major advantage of this method lies in the fact that classification criteria are generated automatically and do not need specific tuning. This is why we decided to adapt this technique to a vehicle classification context. We use our improved Genetic Algorithms based Adaboosting, which we called Adaboost/GA, with the enhanced illumination-independent classifiers described in Khammari et al. (2005). This approach will be briefly described in the next paragraphs.

Boosting consists of linearly combining a set of weak classifiers to obtain a strong one. In our case, we use the weak classifiers we developed and described in Khammari et al. (2005) composed of two sets of control points: $X = \{x_1,\ldots, x_n\}$ and $Y = \{y_1,\ldots, y_m\}$, (n, $m < 6$) and the resolution (48×36, 24×18 or 12×9) on which it operates. An example is shown in Figure 10. The classifier answers 'yes' if one of the following conditions is verified for a given scale:

$$\min_m\left(I\left(y_m\right)\right) > \max_n\left(I\left(x_n\right)\right) \text{ or } \min_n\left(I\left(x_n\right)\right) > \max_m\left(I\left(y_m\right)\right)$$

AdaBoost iteratively takes the best simple classifier it could find at each step, and adds it to its final set of classifiers while updating its coefficient (according to the weight of examples distribution). Note that at each iteration of the algorithm, it tries to find the best classifier for the learning examples which have been least treated so far. Obviously, choosing the best simple classifier at each step of AdaBoost cannot be done by testing all the possibilities. Therefore, a genetic-like algorithm is used (Abramson and Steux, 2004), which starts with a set of random simple classifiers and iteratively improves them while using the following mutations: changing the control points' number and positions, the threshold and the scale.

Figure10 Examples of weak classifiers: the set X is shown in green and the set Y in red (for colours see online version)

The genetic-like algorithm maintains a set of simple classifiers which are initialised as random ones. During each step of the algorithm, a new 'generation' of simple classifiers is produced by randomly applying some mutations (move, add or suppress a control point) on each of the simple classifiers. All of the mutations are tested and the one with the lowest error may replace the 'parent' if it has a lower error. In addition, some random simple classifiers are added at each step.

The strong final classifier, composed of *N* weak ones, will be then used to attribute a classification score between 0 and 1 to each new example. The choice of a value for *N* will be discussed in the next paragraph.

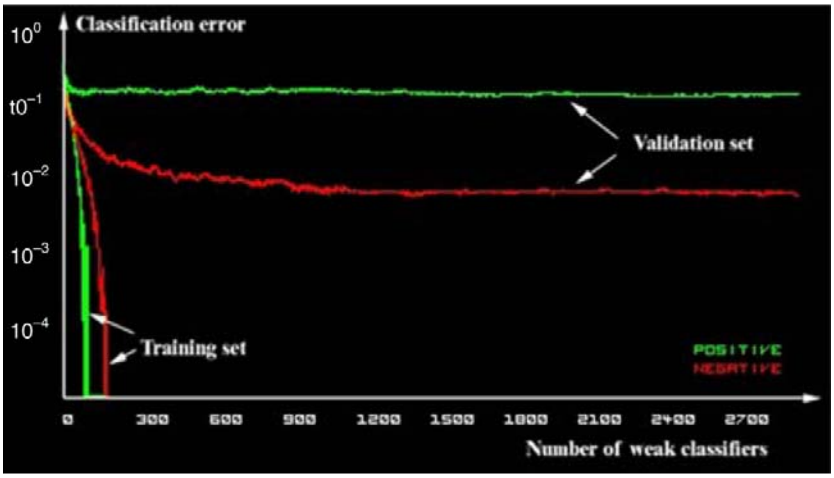
4.2 Off-line learning process

To ensure a good variety of data in each session, the images used in the off-line training were taken on different day times, as well as on different highways and urban scenes. The training set contains subimages of 48 × 36 pixels of rear vehicle views and non-vehicles which were extracted semi-automatically using the Semi- Automatic Visual Learning, (SEVILLE), (Abramson and Freund, 2005). This software offers a method for fast collection of high-quality training data for visual object detection.

We collected a total of 1500 vehicle subimages (positive samples) and 11,000 non-vehicle subimages (negative samples) that were divided into the training set and the validation according to a (2/3, 1/3) proportion. We run AdaBoost/GA with 2000 weak classifiers. As one can see on Figure 11, the classification error in the training set decreases considerably once 150 classifiers are reached, while we need about 1000 classifiers to get an error less than 0.01 in the validation set.

With 1000 classifiers, the false DR is about 0.01. That means that only 1 non-vehicle subimage out of 100 is identified as a vehicle. However, the non-DR is about 0.2, which is still high. This is due to the severity of the classification algorithm which is sensitive to the vehicle position in the subimage. It must be centred and have the right proportions. However, the hypothesis generation step may not verify these constraints if the gradient maxima localisation is not accurate. This difficulty will be overcome in the target validation phase.

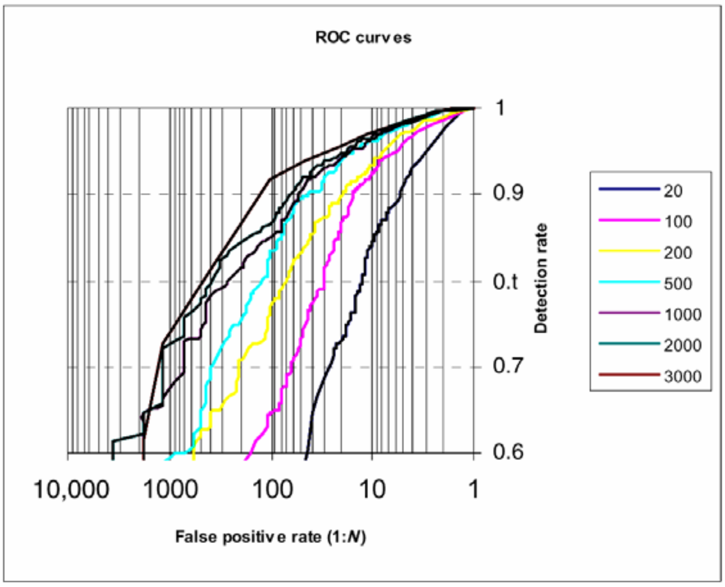
Figure 11 Classification error as a function of the number of weak classifiers (threshold score = 0.5) (for colours see online version)



Some ROC curves for different numbers of classifiers are shown in Figure 12. For instance, with 3000 classifiers, in order to get a DR of about 90%, we must tolerate that one non-vehicle over 100 will be misclassified.

We can see that N = 500 classifiers seems to offer a good compromise between DR, FAR and computation time. In fact, the higher N, the more consuming the module will be. In the next paragraph we will present the two main algorithms based on the final classifier: the validation and the tracking processes.

Figure 12 ROC curves for different classifiers numbers (for colours see online version)



4.3 AdaBoost validation

Once the offline training is achieved, we get a set of satisfactory classifiers that we use to classify at each frame, the new targets given by the hypothesis generation step. Each ROI is resized to match the training subimages then its AdaBoost score is calculated and considered as the confidence of the classification algorithm. Hence, each new target is classified only once. This considerably reduces the computation time.

4.4 AdaBoost tracking

Firstly, the goal of this step is above all to eliminate non-detections that could occur when the hypothesis generation module fails. Secondly, it should be able to identify the same vehicle overtime to evaluate its distance and relative speed if needed. We developed an approach similar to Support Vector Tracking (SVT) technique, described in Avidan (2001) in which we substituted SVM by AdaBoost.

For each lost target from the $t - 1$ frame, we check its surroundings to find a subimage that gives the highest classification score which will be considered as the confidence of the tracking algorithm and hence used for fusion later. The search is exhaustively performed in a surrounding that is proportional to the lost ROI's height and width. One can note that the estimation of the target movement will help limit the search space and hence accelerate the tracking.

5 Fusion framework

Let us remind that the target generation step maximises the DR while the validation step tends to minimise the FAR but also affects the DR. Hence, using them sequentially will give a poor detection performance. Besides, we should take into account uncertainty, inaccuracy and the reliability over the data delivered by our physical and logical sensors (algorithms) and be able to handle redundancy or conflict situations. For this reason data fusion techniques are inevitable.

Data fusion is often used in literature for estimation purposes (Kalman filtering, particle filtering, etc.) and hence model the uncertainty and inaccuracy of, for instance, the distance and velocity of a given target (Arulampalam et al., 2002). This estimation could further be used to validate the targets. The approach used here is different, since we focus on the classification task to decide whether the target is a vehicle or not. If it is the case then we will proceed to estimation. In fact, we want to be sure that a given target is a vehicle before estimating its distance and velocity. For this sake, only the confidence level will be addressed. Besides, we will operate at the symbolic level, which means that we will fuse target candidates rather than raw or mid-processed data.

Then three questions should be addressed:

1 How do we model uncertainty?

2 How should we combine confidence levels issued from each sensor and algorithm?

3 How to drive the final decision: vehicle or not?

The answer to these questions could be found in three main theoretical frameworks: probabilities (Li et al., 2002; Srinivass et al., 2003), possibilities (Dubois and Prade, 2000) and beliefs (Shafer, 1976) (or theory of evidence). We have chosen the latter framework because it is a generalisation of the first two in the discrete events, which is the case here. Moreover, this framework can handle efficiently ignorance and does not need any a priori knowledge about events distributions, models, etc. which are not usually available in practice. Besides, like the other frameworks, it offers a practical way to deal with sources (sensors) reliability.

Two main models are used in the theory of evidence: Inaccurate Probabilities (Kohlas and Monney, 1995; Walley, 1995) and the TBM (Smets and Kennes, 1994). The first model tries to envelop the right probability function with an upper and a lower envelope. The TBM offers a subjective representation of uncertainty based on belief functions instead of the probabilities and contains a set of very practical tools for representing and combining uncertainty and for decision making. The two major ones used for classification are the General Bayes Theorem (a generalisation of the Bayes Theorem to belief functions) and the pignistic transformation. The following section gives a brief overview of this model inspired from Delmotte and Smets (2004) to which one can refer for more details.

5.1 The transferable belief model

This model, which represents an interpretation of the Shafer's model (Shafer, 1976), was introduced by Smets (1994) and mainly developed by Delmotte and Smets (2004). Let us denote by Ω, the hypothesis domain which is in our case: $\{V, \mathrm{NV}\}$, V for Vehicle and NV for Non-Vehicle. And let A represent a set of hypothesis from Ω.

The central element of the TBM is the basic belief assignment (bba), denoted m. For A in Ω, $m(A)$ is the part of belief that supports A (i.e. the fact that the actual value H_0 of is in A), and that, due to a lack of information, does not support any strict subset of A. The initial total belief is scaled to 1, and thus:

$$m(A) \in [0,1] \text{et} \sum_{A \in \Omega} m(A) = 1$$

Let us denote by α, the reliability of a given source S. Then, $m[S](A) = \alpha m(A)$. One can note that it is a simple multiplication that, once performed we can consider the given source as fully reliable.

The degree of belief bel(A) is defined as: bel: $2^{\Omega} \to [0, 1]$, with for all A in Ω:

$$\text{bel}(A) = \sum_{\varnothing = B \subseteq A} m(B)$$

It quantifies the total amount of 'justified specific' support given to A. The term 'justified' means that supports, thus B is in A, and the term 'specific' means that B does not support $\overline{A}$.

The degree of plausibility is defined as: pl: $2^{\Omega} \to [0, 1]$, with, for all A in Ω:

$$\text{pl}(A) = \sum_{A \cap B \neq \varnothing} m(B)$$

It quantifies the maximum amount of 'potential specific' support that could be given to A. The term 'potential' means that B might come to support A without supporting $\overline{A}$ if a further piece of evidence is taken into consideration, thus $B \cap A \neq \varnothing$.

The commonality function is defined as: q: $2^{\Omega} \to [0, 1]$, with, for all A in Ω:

$$q(A) = \sum_{B: A \subseteq B} m(B)$$

The functions m, bel, pl and q are always in one to one correspondence. They all describe the same information but seen under different points of view.

Conjunctive combination of belief functions, which assumes that the sources are fully reliable, is simply obtained by multiplying the commonality function of each source. We can then deduce the other functions, degrees of belief and plausibility, from equations given in Delmotte and Smets (2004).

The General Bayesian Theorem extends the probabilistic one to the belief functions. Let's denote by $l(H/s)$, the likelihood of the hypothesis H given the measure s from a sensor S. In our case, the degree of plausibility will be considered as the likelihood function. Then, the bba of an event A in Ω, given the observation s is obtained with this equation:

$$m^{\Omega}[s](A) = \prod_{H \in A} l(H \mid s) \prod_{H \in \overline{A}} (1 - l(H \mid s))$$

In order to decide, we have to switch to probabilistic space. This is generally done with the pignistic transformation given by:

$$\text{BetP}(H) = \sum_{A: H \in A \subseteq \Omega} \frac{m(A)}{|A|(1 - m(\varnothing))}, \forall H \subseteq \Omega$$

5.2 Targets identification using TBM

First of all, we have to define bba for each hypothesis. For each physical or logical sensor, the confidence itself will be considered as $m(V)$. While $m(\mathrm{NV})$ is initialised to $1 - m(V)$, considering that initially $m(\Omega) = 0$. For each source, a normalised coefficient of reliability of detection and non-detection will be given. These coefficients are chosen based on the experience and the validation we have of each algorithm or sensor and are summarised in Figure 13. Thus the derivation of the coefficients is obtained by the comparison of the result data obtained from our algorithms with the true data registered in ground-truth databases obtained thanks to an important ground truth generation campaign using experimental data described in Section 6 (Table 1).

Figure 13 Reliabilities table

	V	NV
Vision	70 %	70 %
Laser	70 %	95 %
AdaBoost	97 %	80 %

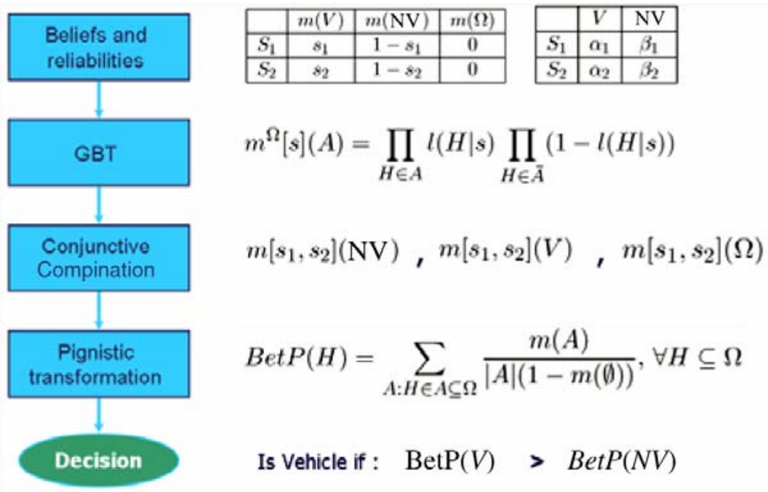
Table 1 Qualitative results in ARCOS

Highways roads (4 lanes)	186 km
National roads (2 lanes)	55 km
Country/secondary roads	72 km
Urban roads	10
PC/sensor breakdown	16 km
Number of false detections	4
Number of non-detections	4
Total	339 km

One can notice that these coefficients could be defined online by supervising for example lightening, weather, traffic conditions, etc. that are likely to impact the result of the detection.

Figure 14 summarises the whole fusion process for two sources. The final decision will be drawn by comparing the pignistic probabilities of the different events: BetP(V), BetP(NV) and BetP(Ω), which in our case comes to compare their corresponding bba: $m(V)$, $m(\mathrm{NV})$ and $m(\Omega)$. Remember that $m(\Omega)$ reflects the ignorance about the real nature of the target to classify, therefore it is a key value for the decision-making process.

Figure 14 Fusion process for two sources



6 Experiments and results

The prototype vehicle we used for this application is equipped with long range radar, a 2D laser scanner, four digital colour CCD cameras, a Trimble DGPS receiver, a FOG Crossbow inertial sensor and odometers. Vehicle information is transmitted via a CAN bus. We also use a Navteq GIS for map matching and geolocalisation. All sensor information is synchronised using the RTMAPS[1] system which is a real-time framework for prototyping multisensor automotive applications (Nashashibi et al., 2001). This system was developed in our laboratory and is currently installed in the prototype vehicle.

The video stream was acquired from the frontal camera mounted near the rear-view mirror with a 50° horizontal field of view and a dynamic range of 50 dB.

In order to evaluate the performance of our detection system, tests were conducted under both simulation and real world conditions. Using RTMAPS, we recorded different scenarios including highway, rural and urban scenes at different times of day. Figure 15 shows various conditions under which real time representative tests were conducted.

For each sequence, we evaluated the DR (Table 2) and the FAR (Table 3) rates. We also compared those rates for each perception system (lidar and vision) as well as for the vision-based combined hypothesis generation plus hypothesis validation system (i.e. improved gradient + AdaBoost/GA). Finally, the last compared system was the final global fusion system as shown in Tables 2 and 3.

Many observations can be made accounting these results. The most important one clearly affirms the superiority of the technique based on the global fusion.

We also installed the system on our host vehicle and conducted real-time tests. We were able to achieve, for a 100 km/h speed, a frame rate of approximately 10 frames per second using a standard PC machine (Bi-Xeon 1 GHz and 1 GB of RAM) and without performing specific software optimisations. This system was also experimented in the context of the ARCOS project on different ACC scenarios and demonstrated very good results. Indeed, a specific evaluation campaign was dedicated to our system. It showed the results of Table 2 on the 339 traversed kilometres. In this campaign, neutral evaluators raised the following parameters: the moment of the detection and the distance to the target, the moment and distance of the target loss, the number of detected targets with respect to real existing targets, number of lanes and road characteristics (slope, curvature, cant, etc.), the type of detected/non-detected targets, the road type (highway, urban, etc.), time of day, lightening and climatic conditions, the duration of the detection with respect to the true presence of the target and finally, the detection quality in several conditions such as overtaking, insertions, etc.

Figure 15 Evaluation sequences in different environments: (a) the GIAT track at Satory near Versailles and (b) on Paris' circular Ring

(a) (b)

Table 2 DRs for different perception systems

	Sequence A (%)	*Sequence B (%)*	*Sequence C (%)*
Vision HG	95.1	91.2	93.4
Lidar HG	100	97.2	99.5
Vision HG+HV	90.3	87.4	89.2
Fusion	100	98.9	99.7

Table 3 False alarms rates for different perception systems

	Sequence A (%)	*Sequence B (%)*	*Sequence C (%)*
Vision HG	95.1	91.2	93.4
Lidar HG	100	97.2	99.5
Vision HG+HV	90.3	87.4	89.2
Fusion	100	98.9	99.7

There are no parameters to tune in our system. It is completely autonomous, which makes it suitable for hardware implementation that could make it less time consuming and suitable for a serial production product.

In Figure 16 we illustrate some results obtained by our system on highways and in urban scenes. The numbers in green (which are not legible in the figure) indicate the level of confidence obtained after the fusion. Figure 17 shows the MMI interface developed for the ACC risk assessment system of the ARCOS project.

Figure 16 Qualitative results in different environments (for colours see online version)

Figure 17 Real time execution in the ARCOS experiments: risk assessment for ACC applications

7 Conclusion and future work

We have presented a multisensor perception system that uses a single frontal camera and a laser scanner for vehicle detection and recognition. Multisensor data processing is part of a more global architecture dedicated to data fusion. We designed a generic architecture that proved its robustness and flexibility. This architecture tolerates any change of sensor or processing algorithms provided that they provide the adequate outputs needed for the generic fusion module (targets and levels of confidence) and that a corresponding reliability table is provided for each algorithm. In this paper, we first described very robust image processing algorithms we use to detect and recognise vehicles on the road. These algorithms combine for the first time feature-based algorithms and learning-based algorithms using a genetic guided AdaBoost algorithm; one major result is the maximisation of the DR in a first step then minimising the FAR thanks to AdaBoost classification. Range data are also processed to extract road obstacles and deliver distances and speeds of the lane target candidates. We also discussed the fusion framework we used to combine both heterogeneous data (images and range data) taking into account uncertainty, inaccuracy and the reliability over the data delivered by the sensors and the algorithms; this framework is based on the TBM, which we proved to be the best suited for our application. However, a major issue is to have the reliabilities table which is not always available. One way to afford this table is through creating ground-truth reference data and comparing them to detection algorithms results. We are currently working on the design of a semi-automatic system capable of generating ground-truth and benchmark data from video and/or range data databases.

The excellent experimental framework that was the French national project ARCOS allowed us to prove the viability and the robustness of our system through the development of a robust ACC system. Experimental results show that this system is capable of detecting vehicles, except very distant vehicles, up to 100 m. It works in real-time conditions and can achieve high reliability target detection with low false positive rate in demanding situations such as complex urban environments.

For future works, we plan to explore in detail the influence of camera characteristics on detection results to find the ideal ‘camera’ to use. Meanwhile we will also continue increasing our training set to improve the AdaBoost classification. In this article, only vehicles were considered, but the object domain can be easily extended to trucks, motorcycles, etc. Also other sets of classifiers will be tested and their performance compared to the current ones. Furthermore, we will focus on ACC application needs and try to make exhaustive and comprehensive ground-truth reference databases including information such as precise obstacles, and locations and velocities.

References

Abramson, Y. and Steux, B. (2004) ‘Hardware-friendly detection of pedestrians from an on-board camera’, *IEEE Intelligent Vehicles Symposium*, Parma, Italy, June.

Abramson, Y. and Freund, Y. (2005) ‘Semi-automatic visual learning (SEVILLE): tutorial on active learning for visual object recognition’, *Computer Vision and Pattern Recognition CVPR’05*, San-Diego, California, 5 June.

Ammoun, S., Abuhadrous, I., Nashashibi, F., Laurgeau, C. and Goulette, F. (2004) 'Modélisation 3D d'environnements urbains et routiers numérisés par télémétrie laser embarqué', *Proceedings of the 9th International rendez-vous for 3D digitisation and modelling professionals*, Paris, France, April, pp.28–29 (in French).

Arulampalam, S., Maskell, S., Gordon, N. and Clapp, T. (2002) 'A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking', *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, pp.174–188.

Avidan, S. (2001) 'Support vector tracking', *Computer Vision and Pattern Recognition CVPR'01*, December.

Avidan, S. (2003) 'Subset selection for efficient SVM tracking', *International Conference Computer Vision and Pattern Recognition*, June.

Baltzakis, H., Argyros, A. and Trahanias, P. (2003) 'Fusion of laser and visual data for robot motion planning and collision avoidance', *MVA (15) Journal*, No. 2, pp.92–100.

Delmotte, F. and Smets, P. (2004) 'Target identification based on the transferable belief model interpretation of Dempster-Shafer model', *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 34, pp.457–471.

Dubois, D. and Prade, H. (2000) 'Possibility theory in information fusion', *Proceedings of the 3rd International Conference on Information Fusion*, Vol. 1, July, pp.6–19.

Fang, Y., Masaki, I. and Horn, B. (2001) 'Distance range based segmentation in intelligent transportation systems: fusion of radar and binocular stereo', *Proceedings of the IEEE Intelligent Vehicles Symposium*, Tokyo, Japan, June.

Fanping, B. and Ching-Yao, C. (2005) 'Pedestrian detection in transit bus application: sensing technologies and safety solution, California PATH Program', *Proceedings of IEEE International Conference on Intelligent Vehicles (IV'05).*

Freund, Y. and Schapire, R. (1997) 'A decision-theoretic generalization of on-line learning and an application to boosting', *Journal of Computer and System Sciences*, Vol. 55, No. 1, pp.119–139.

Früh, C. and Zakhor, A. (2001) 'Fast 3d model generation in urban environment', *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Baden-Baden, Germany, August.

Gavrila, D., Kunert, M. and Lages, U. (2001) 'A multi-sensor approach for the protection of vulnerable traffic participants – the PROTECTOR project', *IEEE Instrumentation and Measurement Technology Conference*, Budapest, Hungary, May.

Gern, A., Franke, U. and Levi, P. (2000) 'Advanced lane recognition: fusing vision and radar', *Proceedings of the IEEE Intelligent Vehicles Symposium.*

Hancock, J., Hebert, M. and Thorpe, C. (1998) 'Laser intensity-based obstacle detection', *Proceedings of the IEEE Conference on Intelligent Robots and Systems (IROS).*

Kalinke, T., Tzomakas, C. and Von Seelen, W. (1998) 'A texture-based object detection and an adaptive model-based classification', *IEEE Intelligent Vehicles Symposium'98*, Stuttgart, Germany, October, pp.341–346.

Khammari, A. (2006) 'On-board multi-sensor vehicle detection system: application to ACC', PhD Thesis, École des Mines de Paris, February (in French).

Khammari, A., Nashashibi, F., Abramson, Y. and Laurgeau, C. (2005) 'Vehicle detection combining gradient analysis and AdaBoost classification', *Proceedings of the 8th IEEE Conference on Intelligent Transportation Systems*, September, pp.1084–1089.

Kohlas, J. and Monney, P.A. (1995) 'A mathematical theory of hints: an approach to Dempster-Shafer theory of evidence', *Lectures Notes in Economics and Mathematical Systems*, Berlin, Germany: Springer-Verlag, Vol. 425.

Langheim, J., Buchanan, A., Lages, U. and Wahl, M. (2001) 'CARSENSE-new environment sensing for advanced driver assistance systems', *International Conference of Intelligent Transportation Systems (ITSC 2001)*, Vol. 3, No. 2, Torino, Italy, April, pp.796–801.

Li, B., Wang, R. and Guo, K. (2002) 'Vehicle detection based on distributed sensor decision fusion', *International IEEE Proceedings of ITSC 2002*, Singapore, pp.242–247.

Mendes, A.B. and Nunes, L.C. (2004) 'Multi-target detection and tracking with a laser scanner', *Proceedings of the IEEE Intelligent Vehicles Symposium (IVS'04)*, pp.796–801.

Moutarlier, P. and Chatila, R. (1989) 'Stochastic multi-sensory data fusion for mobile robot location and environment modeling', *Fifth International Symposium on Robotics Research*, pp.207–216.

Nashashibi, F. and Devy, M. (1993) '3D incremental modeling and robot localization using a laser range finder', *IEEE International Conference Robotics and Automation ICRA'93*, Atlanta, 2–7 May, pp.20–27.

Nashashibi, F., et al. (2001) 'RT-MAPS: a framework for prototyping automotive multi-sensor applications', *Mobile Robots*, Vol. 8, No. 2, pp.520–531.

Newman, P. (1999) 'On the structure and solution of the simultaneous localisation and map building problem', PhD thesis, Australian Centre for Field Robotics – The University of Sydney, March.

Shafer, G. (1976) *A Mathematical Theory of Evidence*, Vol. 2702, Princeton University Press.

Smets, P. and Kennes, R. (1994) 'The transferable belief model', *Artificial Intelligence*, Vol. 66, pp.191–234.

Srinivass, N., Chen, Y. and Daniell, C. (2003) 'A fusion system for real-time forward collision warning in automobiles', *IEEE Proceedings of ITS*, pp.457–462.

Stein, G.P., Mano, O. and Shashua, A. (2003) 'Vision-based ACC with a single camera: bounds on range and range rate accuracy', *Proceedings of the IEEE Intelligent Vehicles Symposium*, June.

Steux, B. (2001) '[RT]Maps un environnement logiciel dédié à la conception d'applications embarquées temps-réel. Utilisation pour la détection automatique de véhicules par fusion radar/vision', PhD Thesis, École des Mines de Paris, December (in French).

Sun, Z., Miller, R., Bebis, G. and Dimeo, D. (2000) 'A real-time precrash vehicle detection', *IEEE Intelligent Vehicles Symposium 2000*, Dearborn, MI, USA.

Sun, Z., Miller, R., Bebis, G. and DiMeo, D. (2002) 'A real-time precrash vehicle detection system', *Proceedings of the 2002 IEEE Workshop on Applications of Computer Vision*, Orlando, FL, December.

Trassoudaine, L., et al. (2004) 'Détection d'obstacles par fusion multisensorielle', *ARCOS Deliverable N°R9*, September (in French).

Victorino et al. (2003) 'Safe navigation for indoor mobile robots', *Part I: A Sensor-Based Navigation Framework. The International Journal of Robotics Research*, pp.1005–1118.

Viola, P. and Jones, M. (2001) 'Rapid object detection using a boosted cascade of simple features', *International Conference on Computer Vision and Pattern Recognition CVPR'01*, Dearborn, MI, USA.

Walley, P. (1995) *Statistical Reasoning with Imprecise Probabilities*, London; Chapman and Hall.

Ye, C. and Borenstein, J. (2002) 'Characterization of a 2-d laser scanner for mobile robot obstacle negotiation', *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, Washington, DC, May, pp.2512–2518.

Zhao, H. and Shibasaki, R. (2003) 'A vehicle-borne urban 3D acquisition system using single-row laser range scanners', *IEEE Transactions on SMC Part B: Cybernetics*, Vol. 33, No. 4, August.

Note

[1RT]M@ps is a product of Intempora Inc.